УДК 68Т50

О.О. Марченко, А.О. Никоненко, Т.В. Россада, Є.А. Мельников
Київський національний університет імені Тараса Шевченка, Україна
вул. Володимирська, 64/13, м. Київ, 01601

МЕТОД МАШИННОГО НАВЧАННЯ ДЛЯ ІДЕНТИФІКАЦІЇ ПАРАФРАЗИ

O.O. Marchenko, A.O. Nykonenko, T.V. Rossada, E.A. Melnikov
Taras Shevchenko National University of Kyiv, Ukraine
Volodymyrska St., 64/13, Kyiv, 01601

ABOUT ONE MACHINE LEARNING METHOD FOR PARAPHRASE IDENTIFICATION

У роботі описаний новий ефективний алгоритм ідентифікації парафрази, розроблений з використанням машинного навчання. Архітектура системи має форму багатошарового класифікатора, де класифікатори нижнього рівня приймають рішення про факт наявності або відсутності парафрази в парах речень, відповідно до їхніх індивідуальних стратегій, а супер-класифікатор верхнього рівня приймає остаточне рішення. Експерименти показали оцінки точності визначення парафрази, співставні з кращими існуючими в світі системами.

Ключові слова: машинне навчання, аналіз природномовних текстів, визначення парафрази.

A new effective algorithm for paraphrase identification has been developed with using machine learning approach. Architecture of the system has a form of multilayer classifier where sub-classifiers of the lower level make decisions about presence or absence of paraphrase in sentences according to their strategies and super-classifier of upper level finds the final solution. Experiments demonstrated precision of paraphrase detection comparable with the best ones state-of-the-art systems.

Keywords: machine learning, natural language text processing, paraphrase identification.

Вступ

Ідентифікація парафрази стала однією з найбільш актуальних задач у комп'ютерній лінгвістиці. Можливо тому, що пошук речень, які за лексичним складом є різними, але мають однакове смислове значення, є дуже подібним до класичної задачі визначення семантики текстів природною мовою.

На сьогодні досягнуто значних успіхів у розробці алгоритмів ідентифікації парафрази. Основні дослідження велися і ведуться в рамках напрямку машинного навчання. Системи, які продемонстрували найкращі показники точності визначення на стандартних корпусах парафраз, використовували такі потужні і ресурсомісткі технології, як Recursive Neural Networks, Convolutional Neural Networks і невід'ємну матричну факторизацію. Крім всієї нетривіальності та певної невизначеності, які неминуче виникають при використанні нейронних мереж, слід зазначити також і алгоритмічну складність невід'ємної факторизації матриць, що робить ці методи проблемними рішеннями для застосування в промислових системах, які працюють у реальному часі.

Перед авторами даної статті стояла задача розробки повномасштабної системи визначення наявності парафрази, яка працювала б онлайн у реальному часі з великими потоками текстової інформації, і це накладало відповідні обмеження на швидкість алгоритму. Тому, як базовий підхід, був обраний стандартний метод опорних векторів (SVM) з розробкою оригінальної багаторівневої структури системи класифікаторів. Основною ідеєю запропонованого підходу є розробка набору класифікаторів нижнього рівня і побудова супер-класифікатора, який на основі набору рішень, отриманих від

класифікаторів нижнього рівня, вчиться приймати остаточне рішення про наявність чи відсутність парафрази.

На першому етапі кожен з класифікаторів нижнього рівня навчається якісно розпізнавати деякі типи випадків парафрази/непарафрази за навчальною вибіркою. Для цього навчальна вибірка модифікується під кожний окремий класифікатор нижнього рівня видаленням зайвих навчальних пар речень, які представляють певний «шум» для нього, так як вони, наприклад, не входять у цільову вибірку типів парафрази для даного підкласифікатора, при цьому будучи парафразою, тобто повинні увійти до вибірки іншого відповідного підкласифікатора. Після навчання класифікаторів нижнього рівня йде етап навчання суперкласифікатора. Навчені класифікатори нижнього рівня відпрацьовують весь навчальний корпус. Їх оцінки на всій навчальній множині пар речень є навчальною вибіркою для суперкласифікатора.

Система, що була розроблена та протестована на стандартних корпусах Microsoft Research Paraphrase Corpus (MSRP) [1] і Plagiarism Detection Corpus (PAN) [2], продемонструвала точність визначення парафрази, співставну з кращими відомими системами state-of-the-art.

Системи аналізу парафрази

Більшість попередніх робіт, присвячених визначенню парафрази з використанням методів машинного навчання, зосереджувалися на побудові оптимального набору ознак, тобто на побудові ефективного ознакового простору.

Були винайдені декілька типів ознак, у тому числі:

(1) ознаки на основі рядків, включаючи перетини n-грам як слів, так і символів [3] та ознаки, основані на метриках оцінки якості машинного перекладу [4];

(2) ознаки, основані на знаннях, що використовують зовнішні лексичні ресурси, такі як WordNet [5];

(3) ознаки на основі синтаксису, які обчислюють міри відмінності синтаксичних залежностей у двох реченнях [6];

(4) міри, що обчислюються на корпусах на основі моделей розподілу, подібно до латентного семантичного аналізу [7, 8].

У новітніх роботах дослідники відійшли від «ручного підбору» ознак до моделювання представлень розподілу та нейромережевих рішень. Hua He, Kevin Gimpel та Jimmy Lin [9] використали згорткову нейронну мережу для обчислення мульти-перспективної подібності речень і їх система продемонструвала оцінки точності на рівні state-of-the-art. Cheng та Kartsaklis [10] застосували розподіли разом із рекурсивною нейронною мережею при розробці синтаксичної прив’язки багатозначних слів для глибокої композиційної моделі значень, та перевершили попередній результат. На сьогодні найкращий результат належить дослідникам Ji та Eisenstein [11], які застосовують для оптимізації ознакового простору невід’ємну матричну факторизацію та відстань Кульбака – Лейблера.

Nitin Madnani, Joel Tetreault та Martin Chodorow [12] розробили алгоритм, який складається із восьми метрик якості машинного перекладу, які обчислюють близькість речень, та класифікатора верхнього рівня, що знаходить кінцевий розв’язок на основі значень оцінок метрик нижнього рівня. Незважаючи на відсутність потужних та ресурсоємних обчислень, цей алгоритм демонструє результати рівня state-of-the-art, набагато переважаючі вищезгадані методи по швидкості та по простоті реалізації.

Запропонований у даній статті алгоритм, у деякому сенсі, можна віднести саме до цього класу методів визначення парафрази.

Опис методу

Для визначення наявності парафрази побудовано класифікатор, який має дворівневу структуру. На нижньому рівні вхідні дані, що представляють собою пари речень, які потрібно перевірити, чи не є вони парафразом один до одного, аналізується набором простих класифікаторів, кожен з яких навчений розпізнавати парафрази певного набору типів. Ці класифікатори визначають для кожної вхідної пари наявність або відсутність парафрази. На верхньому рівні отримані результати оцінюються головним класифікатором, який і приймає остаточне рішення.

Для навчання системи необхідна розмічена вибірка пар речень (1 − парафраза/0 − непарафраза) (див. табл.1).

Таблиця 1. Приклад елементів навчальної вибірки

Мітка	Перше речення	Друге речення
1	Amrozi accused his brother, whom he called "the witness", of deliberately distorting his evidence.	Referring to him as only "the witness", Amrozi accused his brother of deliberately distorting his evidence.
0	Yucaipa owned Dominick's before selling the chain to Safeway in 1998 for $2.5 billion.	Yucaipa bought Dominick's in 1995 for $693 million and sold it to Safeway for $1.8 billion in 1998.

Для навчання системи застосовуються ознаки:

1. Sentence Length Difference – порівняння кількості лексем у реченнях:

$$SentenceLengthDifference(r,c)=\frac{|r|-|c|}{|r|},\quad SentenceLengthDifference*(r,c)=\frac{1}{d^{|r|-|c|}}$$

2. N-Grams Comparing – порівняння уніграм, біграм та триграм:

$$NGramsComparing_N(r,c)=\frac{|NGrams_{N,r}\cap NGrams_{N,c}|}{|NGrams_{N,r}|},$$

де $NGrams_{N,x}$ – множина послідовностей слів довжиною N у реченні x.

4. Dependencies Similarity – подібність синтаксичних залежностей

$$DependeciesSimilarity(r,c)=\frac{\sum_{d\in DT_r}\sum_{i\in r|_{dep=d}}\max_{j\in c|_{dep=d}} similarity(i,j)\cdot BP(|r_d|,|c_d|)}{|DT_r|BP(|DT_r|,|DT_c|)},$$

де DT_x – множина усіх наявних синтаксичних залежностей у реченні x; $x|_{dep=d}$ – усі лексеми речення x, що зв'язані відношенням d; $similarity(x,y)$ – числовий показник подібності двох лексем x та y, обчислений на основі бази WordNet; $BP(x,y)=\begin{cases}1, & y>x\\ e^{1-\frac{x}{y}} & \end{cases}$ – *Brevity Penalty*

5. Dependencies Comparing – порівняння синтаксичних залежностей

$$DependenciesComparing(r,c)=\frac{|dependencies(r)\cap dependencies(c)|}{|dependencies(r)|}$$

$$dependencies(x)=\{(i,j,d)\,|\,d\in DT_x, i\in x, j\in x|_{dep=d}\}$$

6. Syntactic N-Grams Comparing − порівняння синтаксичних уніграм, біграм та триграм. Обчислення відбувається так само, як і для звичайних N-грам, але під

синтаксичними N-грамами розуміється послідовність лексем, які є зв'язним підграфом синтаксичного дерева підпорядкування речення.

Також реалізовано міри семантичної близькості, що були розроблені у моделях машинного перекладу.

7. BLEU [13]
8. BLEU, в якому за N-грам беруться послідовності значущих слів, тобто таких, що:

$$IDF(x, docs) < L,$$

де IDF – Inversed Document Frequency: $IDF(x, docs) = \log\left(\frac{|docs|}{|docs|_{x \in docs}}\right)$; $corpus$ – корпус документів, на якому обчислюється IDF; L – деяке граничне значення, що залежить від особливостей корпусу документів.

9. BLEU, у якому за N-грам беруться послідовності синтаксичних N-грам
10. NIST [14]
11. Meteor [15]
12. Badger [16]

Кожна міра $M(x,y)$ має два варіанти реалізації: $precision : M(x, y)$, $recall : M(y, x)$. В усіх мірах, де використовується порівняння лексем, реалізовано по два варіанти: повне співпадіння лем та співпадіння за синонімами.

Частину ознак було отримано із застосуванням принципів семантичної схожості − зв'язності, описаних у [17].

Для навчання класифікаторів нижнього рівня необхідно сформувати навчальні вибірки для кожного з них. Кожен класифікатор має спеціалізуватися на розпізнаванні парафраз певного набору типів. Причому кожен тип парафраз має входити у навчальну вибірку одразу для декількох класифікаторів. Саме цим гарантується надійне покриття всіх типів парафрази та взаємна підстраховка класифікаторів при розв'язанні задачі. Постає питання про те, що таке тип парафрази, як його можна промоделювати та визначити тип парафрази для кожної конкретної пари речень. За робочу гіпотезу було прийняте припущення, що дві пари речень входять до множини парафраз одного типу, якщо існує певна значна підмножина ознак, що мають подібні значення при обчисленні на цих двох парах речень.

На *першому етапі* навчання алгоритму обчислюється матриця train значень ознак f_i для кожної пари речень з навчального корпусу Microsoft Research Paraphrase Corpus (MSRP).

На *другому етапі* матриця train, з використанням алгоритмів кластеризації розбивається на множину матриць: $train_1^+$, $train_2^+, \ldots, train_n^+$ та $train_1^-$, $train_2^-, \ldots, train_m^-$ таких, що:

$$train^+ = \bigcup_{i=1}^{n} train_i^+, \; train^- = \bigcup_{i=1}^{m} train_i^-, train^+ \cup train^- = train,$$

де $train^+$ та $train^-$ – множини усіх векторів значень ознак для пар речень, помічених як парафрази та непарафрази відповідно.

Для того, щоб класифікатори, навчені на вибірках, сформованих на основі цих матриць, могли «підстраховувати» один одного при прийнятті рішення по кожному окремому випадку, кожна вибірка має складатися із пар різного типу парафрази/непарафрази, тобто щоб вибірки інтенсивно перетиналися. Для цього, для $train^+$ та $train^-$ має виконуватись наступна умова (для $s \in \{+,-\}$):

$$\forall i \exists j : train_i^s \cap train_j^s \neq \varnothing$$

Третій етап. На основі отриманих на другому етапі матриць формуються навчальні вибірки шляхом формування $n = C_{N'}^k$ комбінаторних сполучень по множині матриць$\{train_i^+\}$ та доповнення кожної комбінації повним набором із $train^-$. Усі відповідні пари речень, чиї вектори значень ознак увійшли до деякої комбінації, формують навчальну вибірку для деякого класифікатора першого рівня clf_i^+. Таким чином, отримаємо навчальні вибірки для $n = C_{N'}^k$ класифікаторів clf_i^+. Аналогічно формуються навчальні вибірки для $m = C_{M'}^k$ класифікаторів clf_i^-.

Четвертий етап. На основі отриманих вибірок відбувається навчання множини класифікаторів clf_1, clf_2, …, clf_{n+m}. Після навчання класифікатори разом обробляють весь навчальний корпус train. Матриця розв'язків класифікаторів clf_1, clf_2, …, clf_{n+m} для всіх пар речень корпусу train разом з розміткою пар служить навчальною вибіркою для суперкласифікатора верхнього рівня.

На *п'ятому етапі* відбувається навчання класифікатора верхнього рівня.

Для побудови класифікаторів верхнього та нижнього рівня використовується метод опорних векторів.

Алгоритм побудови системи класифікаторів

Крок 1. Автоматичне розбиття $train^+$ на класи за типами парафрази

Пари-парафрази одного типу мають корелювати за значеннями ознак, що обчислюються на них. Тобто, значення повинні бути *близькими.* Деякі ознаки можуть бути неактуальними для парафраз даного типу, тому необхідно визначити множину C^+ таких типових підмножин парафраз $\{c\}$, всередині яких будь-які два елементи мають схожі набори значень для деякого встановленого набору ознак. Формально, мають виконуватися наступні умови.

$$\bigcup_{c \subseteq C^+} c = train^+ \quad \bigcap_{c \subseteq C^+} c \neq \varnothing \ \forall c \subseteq C^+ \forall x, y \in c : x \overset{l}{\approx} y,$$

де C^+ – множина отриманих типових підмножин; F – множина реалізованих ознак; $x \overset{l}{\approx} C_i^+ \Leftrightarrow \exists X \subseteq F : \|X\| = l \ \& \ \forall f \in X : |f(x) - f(y)| < \varepsilon$ для заданих l і ε.

Для вирішення поставленної задачі на множині $train^+$ векторів значень ознак пар речень, помічених як парафрази, обчислюються N центроїдів: $C_1^+, C_2^+, ..., C_N^+$ – найбільш віддалених один від одного елементів з $train^+$. Для центроїдів мають виконуватися наступні умови:

$$C_1^+, C_2^+, ..., C_N^+ \in train^+, \quad \sum_{i, j \in 1..N \& i \neq j} dist(C_i, C_j) \to \max,$$

де $dist$ – евклідова відстань між двома елементами.

Після вибору центроїдів кожен елемент з вибірки $train^+$ додається до одного або декількох кластерів, що визначаються центроїдами $C_1^+, C_2^+, ..., C_N^+$: елемент x потрапляє в кластер c_i^+, який визначається центроїдом C_i^+, якщо виконується умова: $x \overset{l}{\approx} C_i^+$. Таким чином будуються N кластерів: $c_1^+, c_2^+, ..., c_N^+$.

Для елементів, що не потрапили в жодний клас, рекурсивно виконується Крок 1, але зі зміненими параметрами N, l и ε. У результаті будується N' кластерів, кожен з

яких складається з набору пар речень-парафраз з навчальної вибірки, які в об'єднанні дають вибірку $train^{+}$, а також мають не порожній перетин. Інтенсивність перетинів визначається початковими параметрами *N, l* і ε.

Крок 2. З $c_1^{+}, c_2^{+}, ..., c_{N'}^{+}$ формуються $n = C_{N'}^{k}$ усіх можливих комбінацій об'єднань *k* кластерів, до кожної з яких додається також весь набір $train^{-}$. У результаті відбору відповідних пар речень з корпусу ***train*** отримано навчальні набори T_1^{+}, T_2^{+}, …, T_n^{+}.

Крок 3. З використанням стандартних методів з бібліотеки sklearn.feature_selection [18] під кожну навчальну вибірку T_1^{+}, T_2^{+}, …, T_n^{+} для методу SVM формується свій власний оптимальний набір ознак $\{f_1, f_2,.., f_k\}$ з початкової множини реалізованих ознак *F*.

Крок 4. Класифікатори нижнього рівня clf_1^{+}, $clf_2^{+}, \ldots, clf_n^{+}$ навчаються на T_1^{+}, T_2^{+}, …, T_n^{+}, використовуючи відповідні оптимальні набори ознак.

Крок 5. Аналогічним чином генеруються класифікатори clf_1^{-}, $clf_2^{-}, \ldots, clf_m^{-}$. Разом з clf_1^{+}, $clf_2^{+}, \ldots, clf_n^{+}$ вони являють собою класифікатори нижнього рівня: clf_1, clf_2, …, clf_{n+m}.

Крок 6. Після навчання класифікатори разом обробляють весь навчальний корпус ***train***. Матриця розв'язків класифікаторів clf_1, clf_2 …, clf_{n+m} для всіх пар речень корпусу train разом з розміткою пар служить навчальною вибіркою для суперкласифікатора верхнього рівня. Виконується навчання класифікатора верхнього рівня.

Крок 7. Разом вся навчена система обробляє тестовий корпус пар речень.

Результати експериментів

Навчання і тестування проводилося на вибірці Microsoft Research Paraphrase Corpus [19]. Корпус складається з 5800 пар речень із різних джерел із зазначенням, чи є пара парафразом, і розділений на навчальну вибірку, що складається з 4076 пар речень (2753 позитивних: 67,5%) і тестову, що складається з 1725 пар речень (1147 позитивних: 66,5%). На сьогодні кращі результати на даній вибірці показали методи [20] з таблиці 2**.**

Таблиця 2. Paraphrase Identification. State of the art

Algorithm	Reference	Description	Supervision	Accuracy	F
MTMETRICS	Madnani et al. (2012)	combination of eight machine translation metrics	supervised	77.4%	84.1%
Multi-Perspective CNN	He et al. (2015)	Multi-perspective Convolutional NNs and structured similarity layer	supervised	78.6%	84.7%
SAMS-RecNN	Cheng and Kartsaklis (2015)	Recursive NNs using syntax-aware multi-sense word embeddings	supervised	78.6%	85.3%
TF-KLD	Ji and Eisenstein (2013)	Matrix factorization with supervised reweighting	supervised	80.4%	85.9%

Для оцінки якості класифікації використані стандартні метрики:

$$F1 = \frac{2 \cdot precision \cdot recall}{precission + recall}, \quad accuracy = \frac{tp + tn}{tp + tn + fp + fn}, \quad precission = \frac{tp}{tp + fp},$$

$$recall = \frac{tp}{tp + fn},$$

де tp – кількість коректно визначених парафразів; tn – кількість коректно визначених непарафразів; fp – кількість неправильно визначених парафраз; fn – кількість неправильно визначених непарафраз.

Для експериментів використовуються бібліотеки scikit-learn (http://scikit-learn.org/stable/).

Таблиця 3. Результати експерименту

№ експерименту	**Класифікатор**				**Precision**	**Recall**	**F1**
	К-сть центроїдів	**Число ознак**	**Епсілон**	**Число k злитих кластерів**			
1	5	7	1.3e-3	3	0.7502	0.9503	0.8384
2	5	8	1.3e-3	2	0.7676	0.9564	0.8516

Як видно з таблиці 3, запропонований метод показав результати, співставні з найкращими існуючими на сьогодні алгоритмами, не використовуючи при цьому таких складних та потужних підходів, як нейронні мережі, латентний семантичний аналіз та невід'ємну факторизацію матриць. Якщо вважати розроблений метод певним продовженням та розвитком алгоритму [12], то по оцінках точності даний метод помітно переважає попередника.

Подяка

Автори статті дуже вдячні компанії P1K, і зокрема команді проекту Unplug, за підтримку в дослідженнях та допомогу в розробці даного методу визначення парафрази, в його тестуванні та впровадженні в продукти компанії.

Висновки

У роботі описано новий ефективний алгоритм ідентифікації парафрази, розроблений з використанням машинного навчання. Експерименти показали оцінки точності визначення парафразу, співставні з кращими існуючими в світі системами.

Література

1. Dolan B., Quirk C., Brockett C. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In Proceedingsofthe20thInternationalConferenceon Computational Linguistics, 2004.
2. Potthast M., Stein B., Barron-Cedeno A., Rosso P. An Evaluation Framework for Plagiarism Detection. In Proceedings of COLING, pp. 997–1005, 2010.
3. Wan S., Dras M., Dale R., Paris C. Using Dependency-based Features to Take the "Para-farce" out of Paraphrase. In Australasian Language Technology Workshop, pp. 131–138, 2006.
4. Madnani N., Tetreault J., Chodorow M. Re-examining machine translation metrics for paraphrase identification. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 182–190, 2012.
5. Fellbaum C. WordNet: An Electronic Lexical Database. MIT Press, 1998.

6. Das D., Smith N.A. Paraphrase identification as probabilistic quasi-synchronous recognition. In Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguisticsand the 4th International Joint Conference on Natural Language Processing of the AFNLP, pp. 468–476, 2009.
7. Hassan S. Measuring Semantic Relatedness Using Salient Encyclopedic Concepts. Ph.D. thesis, University of North Texas, Denton, Texas, USA, 2011.
8. Guo W., Diab M. Modeling sentences in the latent space. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 864–872, 2012.
9. He, Hua, Gimpel K., Lin J. Multi-Perspective Sentence Similarity Modeling with Convolutional Neural Networks, Proceedings of EMNLP 2015, Lisbon, Portugal, pp. 1576-1586.
10. Cheng J., Kartsaklis D. Syntax-Aware Multi-Sense Word Embeddings for Deep Compositional Models of Meaning, Proceedings of EMNLP 2015, Lisbon, Portugal, pp. 1531-1542.
11. Ji Y., Eisenstein J. Discriminative Improvements to Distributional Sentence Similarity, Proceedings of Empirical Methods in Natural Language Processing (EMNLP 2013), Seattle, Washington, USA, pp. 891—896.
12. Madnani N., Tetreault J., Chodorow M. Re-examining Machine Translation Metrics for Paraphrase Identification, Proceedings of 2012 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2012), pp. 182-190.
13. Papineni K., Roukos S., Ward T., Zhu W.J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of ACL, 2002.
14. Doddington G. Automatic Evaluation of Machine Translation Quality using N-gram Co-occurrence Statistics. In Proceedings of HLT, pp. 138–145, 2002.
15. Denkowski M., Lavie M. Extending the METEOR Machine Translation Metric to the Phrase Level. In Proceedings of NAACL, 2010.
16. Parker S. BADGER: A New Machine Translation Metric. In Proceedings of the Workshop on Metrics for Machine Translation at AMTA, 2008.
17. Никоненко А.О. Дослідження статистичної схожості-зв'язності // Вісник КНУ імені Тараса Шевченка, серія фізико-математичні науки. — 2016. — № 1 — С. 131—136.
18. [Електронний ресурс]. – Режим доступу: http://scikit-learn.org/stable/modules/feature_selection.html
19. [Електронний ресурс]. – Режим доступу: https://www.microsoft.com/en-us/download/details.aspx?id=52398
20. [Електронний ресурс]. – Режим доступу: https://www.aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art))

Literatura

1. Dolan B., Quirk C., Brockett C. Unsupervised construction of large paraphrase corpora: exploiting massively parallel news sources. In Proceedingsofthe20thInternationalConferenceon Computational Linguistics, 2004.
2. Potthast M., Stein B., Barron-Cedeno A., Rosso P. An Evaluation Framework for Plagiarism Detection. In Proceedings of COLING, pp. 997–1005, 2010.
3. Wan S., Dras M., Dale R., Paris C. Using Dependency-based Features to Take the "Para-farce" out of Paraphrase. In Australasian Language Technology Workshop, pp. 131–138, 2006.
4. Madnani N., Tetreault J., Chodorow M. Re-examining machine translation metrics for paraphrase identification. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 182–190, 2012.
5. Fellbaum C. WordNet: An Electronic Lexical Database. MIT Press, 1998.
6. Das D., Smith N.A. Paraphrase identification as probabilistic quasi-synchronous recognition. In Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguisticsand the 4th International Joint Conference on Natural Language Processing of the AFNLP, pp. 468–476, 2009.
7. Hassan S. Measuring Semantic Relatedness Using Salient Encyclopedic Concepts. Ph.D. thesis, University of North Texas, Denton, Texas, USA, 2011.
8. Guo W., Diab M. Modeling sentences in the latent space. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pp. 864–872, 2012.
9. He, Hua, Gimpel K., Lin J. Multi-Perspective Sentence Similarity Modeling with Convolutional Neural Networks, Proceedings of EMNLP 2015, Lisbon, Portugal, pp. 1576-1586.
10. Cheng J., Kartsaklis D. Syntax-Aware Multi-Sense Word Embeddings for Deep Compositional Models of Meaning, Proceedings of EMNLP 2015, Lisbon, Portugal, pp. 1531-1542.
11. Ji Y., Eisenstein J. Discriminative Improvements to Distributional Sentence Similarity, Proceedings of Empirical Methods in Natural Language Processing (EMNLP 2013), Seattle, Washington, USA, pp. 891—896.
12. Madnani N., Tetreault J., Chodorow M. Re-examining Machine Translation Metrics for Paraphrase Identification, Proceedings of 2012 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2012), pp. 182-190.

13. Papineni K., Roukos S., Ward T., Zhu W.J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of ACL, 2002.
14. Doddington G. Automatic Evaluation of Machine Translation Quality using N-gram Co-occurrence Statistics. In Proceedings of HLT, pp. 138–145, 2002.
15. Denkowski M., Lavie M. Extending the METEOR Machine Translation Metric to the Phrase Level. In Proceedings of NAACL, 2010.
16. Parker S. BADGER: A New Machine Translation Metric. In Proceedings of the Workshop on Metrics for Machine Translation at AMTA, 2008.
17. Nykonenko A.O. Doslidzhennya statystychnoyi skhozhosti-zv"yaznosti // Visnyk KNU imeni Tarasa Shevchenka, seriya fizyko-matematychni nauky. — 2016. — # 1 — C. 131—136.
18. [Elektronnyy resurs]. – Rezhym dostupu: http://scikit-learn.org/stable/modules/feature_selection.html
19. [Elektronnyy resurs]. – Rezhym dostupu: https://www.microsoft.com/en-us/download/details.aspx?id=52398
20. [Elektronnyy resurs]. – Rezhym dostupu: https://www.aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art)

RESUME

O.O. Marchenko, A.O. Nikonenko, T.V. Rossada, E.A. Melnikov

About one machine learning method for paraphrase identification

A new effective algorithm for paraphrase identification has been developed with using machine-learning approach. Architecture of the system has a form of multilayer classifier where sub-classifiers of the lower level make decisions about presence or absence of paraphrase in sentences according to their strategies and super-classifier of upper level finds the final solution.

In the first phase each lower level classifier is trained to detect certain types of paraphrase / non-paraphrase cases on the special prepared training set. For this purpose the training set is modified for each individual lower level classifier by removing unnecessary training pairs of sentences that represent a "noise" for this classifier, because these pairs, for example, are not included in the target types of paraphrase for this sub-classifier, while being a paraphrase, so these pairs must be included to other sub-classifier training set. After lower-level classifiers learning, the phase of super-classifier training follows. The trained lower-level classifiers process the whole training set. The lower-level classifiers assesses of the whole training set sentences pairs have been used by super classifier as a training set.

The system has been developed and tested on standard Microsoft Research Paraphrase Corpus (MSRP). Experiments demonstrated precision of paraphrase detection comparable with the best state-of-the-art systems.

Надійшла до редакції 14.09.2016

